

# **VOICE APPLICATIONS IN OFFICE AUTOMATION**

A Thesis Submitted  
in Partial Fulfilment of the Requirements  
for the Degree of

**Master of Technology**

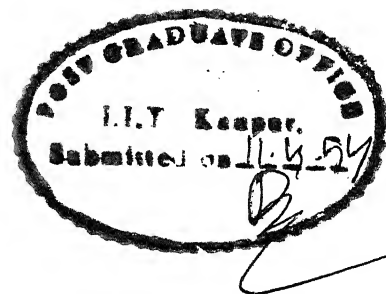
by  
**Sanjay P. Shah**

to the  
DEPARTMENT OF INDUSTRIAL & MANAGEMENT ENGINEERING

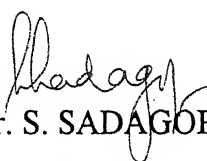
**INDIAN INSTITUTE OF TECHNOLOGY, KANPUR**

APRIL, 1994.

## CERTIFICATE



This is to certify that the present work on "Voice Applications in Office Automation" by Sanjay P. Shah has been carried out under my supervision and has not been submitted elsewhere for the award of a degree.

  
(Dr. S. SADAGOPAN)

Professor

April, 1994

Industrial & Management Engineering

Indian Institute of Technology

Kanpur - 208 016.

21 APR 1994

CENTRAL LIBRARY  
F. I. C. KANPUR

**Acc. No. A. -117712**

711

656-27

SA 187

IME-1994-M-SHA-V01

## ACKNOWLEDGEMENT

With a deep sense of respect and gratitude, I thank Dr. Sadagopan for his valuable guidance and constant encouragement. Without his active interest and support this work would not have been completed. I thank Dr. V. Chaitanya and Dr. Rajiv Sangal of CSE Department for their help and guidance during this work.

I am also thankful to Mrs. Sadagopan for her help and co-operation. I am thankful to Manish Mehta, Subhransu Banerjee, Ashwini and Vijay who inspite of their busy schedule took time to help me. I feel indebted to my classmates Swami, Capt Ramesh for making my stay here enjoyable. I would like to place on record my appreciation for the IME family .

Sanjay Shah.

## CONTENTS

LIST OF FIGURES	vi
ABSTRACT	vii
CHAPTER 1 INTRODUCTION	1
1.1 WHAT IS OFFICE AUTOMATION ?	1
1.2 COMPONENTS OF OFFICE AUTOMATION.	3
1.3 VOICE IN OFFICE AUTOMATION	6
1.4 VOICE	8
1.4.1 Voice characteristics	8
1.4.2 Voice storage and retrieval	9
1.4.3 Digitized sound vs Synthesized sound	11
1.5 ORGANIZATION OF THE THESIS	12
CHAPTER 2 LITERATURE SURVEY	13
2.1 MULTIMEDIA	13
2.2 TEXT-TO-SPEECH SYNTHESIS SYSTEM	15
2.2.1 Text Processing	15
2.2.1.1 Text Analysis	16
2.2.1.2. Phrasing and Intonation	17
2.2.1.3. Letter to Phoneme Conversion	18
2.2.2 Interpretation of Linguistic Structures	18

CHAPTER 3 DESIGN AND IMPLEMENTATION	23
3.1 INTRODUCTION	23
3.2 SCOPE	23
3.3 DESIGN ISSUES	24
3.4 IMPLEMENTATION	30
3.4.1 Introduction	30
3.4.2 Application A: Devanagari TTS	31
3.4.3 Application B: Voice output	34
3.4.4 Application C: Voice Query	35
3.5 LIMITATIONS	36
 CHAPTER 4 CONCLUSIONS AND RECOMMENDATIONS.	 38
4.1 CONCLUSIONS	38
4.2 RECOMMENDATIONS FOR FUTURE WORK.	38
 REFERENCES	 40
APPENDIX	43

## LIST OF FIGURES

FIG	TITLE	PAGE
2.1	TTS Synthesis	15a

## ABSTRACT

This thesis outlines some interesting and innovative applications of voice in the context of office automation in an Indian environment. Three prototype applications have been implemented in this thesis. The first one is an attempt to provide voice output capability to Indian script text. At present the implementation is limited to Devanagiri character; with a minor modification it can be extended to cover other Indian language scripts as well. The second application provides voice capability to any text database. The third application extends the query capability of databases to include voice output as an interesting option. The prototype applications are general in nature and admit dozens of interesting implementations in the financial, services and utility sectors. All the implementations are made using low cost solutions on a PC platform with minimal additional hardware requirements.



## CHAPTER 1

### INTRODUCTION

#### 1.1 WHAT IS OFFICE AUTOMATION ?

The office functions have essentially remained the same - that of information processing. It is only the speed of information processing which has increased tremendously, keeping pace with the demands made on it by the overall modernization of the society in every era. It is the information processing capability, and speed, which determine the effectiveness of an office. The tools of managing information have been mechanized to ensure effectiveness of an office and mechanization has been commensurate with the overall modernization of the times. The office has thus historically evolved.

While it may appear obvious as to what is Information and what is an Office, there is a need to scientifically define both terms. This will help to develop a conception of Office Automation and to put it in proper perspective.

There are numbers of definition about office automation. Olson and Lucas [5] define "Office automation refers to the use of integrated computer and communications system to support administrative procedure in an office environment.". Uhlig [5] defines "An office in which interactive computer tools are put in the hands of individual knowledge workers at their desks in the area in which they are physically working.". Ellis and Nutt [5] define "An automated office information system attempts to perform the functions of the ordinary office by means of a computer system".

In a broad sense, Office Automation is the use of Technology for generation, storage and retrieval , processing and communication of information for improving the effectiveness of office, which in turn will help for realize the objective or business function of the organization in an efficient and competitive manner.

The important terms that need to be understood before we proceed any further are

- Functions of office.
- Nature of work specific to the office.
- Procedures.

Thus within the general function described under "nature of office work", the specifics must be studied and analyzed for the office in question. It is this alone which can throw up areas or weakness that need automation. At the same time, various office automation technologies available need to be studied, and matched with the functions of the office. There can be no simple definition of office automation such as word processing, electronic mail or local area network or use of point-of-sale terminals. Office automation embraces all the relevant information technology for all the activities of an office.

Typically functions of office are accounting, billing, payroll etc. These are automated through the use of computer based system. The office workers performed these functions but they are now part of data processing applications. Much of the processed financial data is now subject of analysis in electronic spread-sheet for decision support functions by professionals and managers.

Today, it is the decision support which also forms part of office automation. The point that becomes obvious is that office automation varies over time, with changes in the activities performed by office workers, and as a result of absorption of technology in the offices.

## **1.2 COMPONENTS OF OFFICE AUTOMATION.**

Office Automation incorporates technology to serve rather than be served by the people, for people are more valuable than the equipment. System planners are now incorporating principles of human factors engineering into the design of hardware as well as software, resulting in user-friendly system.

Except for data processing and photocopying, there was little concentration on office technology prior to the 1980s [5]. Now, however, technology is flooding in the office environment, and with it has come a marked improvement in the design and implementation of tools for information management.

Good office systems are modular and highly flexible, providing users with as much freedom of choice as possible. They offer versatility and transparency of time and place, that is the local time of the day and location of people has little or no impact, even if the people involved are in different time zones or different geographic locations. Personal computers can enable employees to access electronic files and documents in seconds, at any time, from any place. Users of terminals or personal computers can easily and rapidly communicate with a number of people in the same building or on the opposite sides of the earth. Because of office automation, one person can take part in the meetings in different cities in one day without leaving his office.

Virtually the same functional elements tend to recur in the system design of the office automation plans, embracing four modes of information conveyance, i.e. data, text, graphics and audio. These functions are as follows

1. Text Management: Text management means the capturing, manipulation, output, and storage of words and sometimes graphics images. Text management includes electromechanical and electronic typewriters, computer text editors, word processing systems operated on computers. Power and usefulness of word processors far exceed those of typewriters. They provide faster keyboarding and easier formatting of complex documents or text layouts. They also eliminate the need to print information on paper until every portion of it is correct. Nowadays word processing systems equipped with communication capabilities are possessing an immense versatility.

2. Electronic filing and Data Base Management: Electronic filing and retrieving is a rapid and accurate means of accessing information. It is an important component of an Office automation system as Word processors, computer based message systems, electronic calendars and reminders, micrographic units must file (store) and retrieve files of information. Magnetic disks and floppy diskettes are the most important media of storage of the information, but optical disks are making rapid inroads because of their immense, yet, compact storage capabilities.

Database Management Systems contain everything necessary to define, create, maintain, and modify the database both in form and content; and it provides a means of inquiry and report generation. A Query language lets the user quickly write a program to perform queries and to print reports in just minutes.

3. Electronic mail System: An electronic mail system (E-mail) is a point to point conveyer of audio, data/text, graphic modes of information. It can be anything from two telephones to the most advanced computer messaging systems. Today electronic mail systems are either computerized or non-computerized. Many computerized systems consist of terminals organized around network. Electronic mail system provides many advantages like faster delivery of information, less paperwork and photocopying, reduced mailing expenses, geographic independence, improved access to personnel, etc.

4. Teleconferencing: One of the most powerful inventions of office automation is teleconferencing, i.e., meetings without meeting because it provides a way to reduce the vast amount of time as well as the vast amount of money on meetings. Teleconferencing allows people, either individual or groups at distant locations to meet by means of telecommunications. Teleconferencing ranges from a simple long-distance telephone call to a complex electronic integration of audio, video and text. It often lets you have a meeting when time constraints or some situations threaten to delay or cancel the plans. Some of the benefits of the teleconferencing are reduced time and money of travel, ability to attend several meetings at diverse locations in a single day, quicker solutions to the problems and early implementation of the results of meetings.

5. Micrographics: Micrographics is the capture, retrieval and display of miniaturized, high resolution photographic images containing either textual or graphic information. The medium of micrographics is usually film, sometimes paper. Today micrographics is also a valuable part of office automation. It provides for quick and inexpensive duplication of images to be distributed to a group, for easy viewing by projection or computerized display, and for re-creation

of hard copies of microfilms. Micrographics offers compact maintenance of active files and archival storage. With its array of film types, techniques, and retrieval options, micrographics affords the end user many benefits like economy of document and film creation, rapidity of duplication, compact storage, high speed of retrieval and portability. Along with these benefits micrographics offers a few limitations like no space provided for annotation, need to access a viewer terminal, appropriate system for frequent updating, serial accessibility of the film.

### **1.3 VOICE IN OFFICE AUTOMATION**

Speech is one important form of communication in the offices. Its use in the office has not changed over centuries. Various studies have indicated that managers feel more comfortable in communicating through speech, rather than by writing. Their productivity can improve only if office automation provides adequate tools to handle voice. Managers are known to have a certain aversion to keyboard entry either for data entry or retrieval. The utility of an office system is thus limited, however user-friendly it may be. The managers and the professionals can expect to improve their productivity only if their problem areas are addressed, namely, communication and meetings, and decision making activities. For the present, the voice applications in office automation are similar to those of electronic mail. The applications include : Voice message systems, Voice mail systems, Voice reminder systems and Voice answering systems. The basic technology on which the modern voice systems are based is the computer based exchange, microprocessor based auto diallers / telephone instruments, and the concept of digitization of voice for integration of data and voice. The communication network, primarily the telephone network, is an analog transmission system. With the advent of computers and the need to transmit digital data on the analog voice networks, the obvious method was to modulate the

digital two-state signals into electrical analog signal which could be transmitted over the voice network, and demodulate the signal to retrieve the digital data at the receiving end. The modem as a device accomplished this task. The modulation was primarily carried out in the voice bandwidth either by frequency, amplitude, or phase, typically in the range of 110-9600 bps.

Voice MessageMail Store and forward voice message systems are computer based, and feature powerful message handling and management capabilities. These systems operate in a manner similar to the computer based message systems. Simple commands are issued by the user to the voice store and forward computer, using the 12 keys on the touch tone pad of the telephone. The computer interprets the tones generated by the keys. The function of the keys can be easily understood by the user, either through a plastic template fitted over the keypad or through voice prompts, which explain keys that should be used next. A telephone answering machine is the simplest form of a voice message system. The receiver of the messages can control the incoming calls while the sender has no choice but to leave a message or disconnect call. There is no direct conversation of the caller and the called. The spoken message is thus handled as flexibly as the written one. Because of the asynchronous nature of communication, unlike the telephone conversation, no time is lost in exchanging social pleasantries. Time zone differences pose no problem and the same message can be broadcast to more than one recipient. The advantages of voice mail systems are at least as many as in the message systems.

## **1.4 VOICE**

There are many reasons for using voice in office automation. One is that it is possible to communicate emotional cues such as excitement, depression, anger, satisfaction or doubt more easily when using voice than with other media. It is the most natural way to communicate and everyone is comfortable in speaking. Voice is a speedier form of input, the people can dictate vocally up to six times faster than they can write. For many executive, who are not accomplished typists, speech is faster than keyboard entry. In the Indian context voice also has the advantage of its independence from language and script which is multi-lingual.

### **1.4.1 Voice characteristics**

Representing voice in digital form allows the application of digital computer technology to the processing of voice signals. Many techniques exist for voice representation in computer, ranging in complexity from simple sampling and encoding of waveforms to estimation of parameters from human model of speech production. When choosing a particular representation method, the various factors need to be considered. These factors are

1. Processing complexity - which refers to the amount of processing required to obtain a digital representation from analog signals and vice-versa. It is a measure of cost of implementation of the system in hardware and/or software.

2. Data rate - refers to the rate at which digital data has to be transferred from memory to the decoder or synthesizer to reproduce the speech signal. A low data rate will result in the low storage requirements and capability of transferring more information over a channel.



3. Speech quality is a measure of how well the reconstructed signal approximates the original signal. Flexibility is a measure of the degree to which the various attributes of the speech (e.g. pitch, speech rate and quality) can be manipulated independently of one another. It is not always possible to control each attribute independently.

In general any technique should have low processing complexity, low data rate, high speech quality and high flexibility. But these four goals are conflicting and compromises have to be made. Nowadays since cost of hardware is dropping rapidly and special purpose digital signal processors are available, processing complexity is no prime consideration. Instead, storage requirements, despite the falling price of memory are the most important considerations, when system has to represent a large number of utterances. The choice of one set of parameters over another should be made depending upon the application.

#### **1.4.2 Voice storage and retrieval**

Real-life sound is an analog signal, whose pitch("frequency") and loudness("amplitude") varies continuously with time. Computers can store only digital data., i.e. data represented by numbers. The conversion of analog sound to digital is achieved by a process known as "sampling". After every tiny fraction of a second, the sound signal's "dynamic range" is measured, and stored as digital information in bits and bytes. The number of times this is done every second is "sampling rate", and the amount of digital information we record for each sample is our "sample size".

These two parameters bear a very close relation to the quality of output we get. We know that the sound signal keeps changing continuously. With our

sampling technique, we take a "snapshot" of this signal and assume that sound remained static at this level till the next sample was taken. That assumption isn't quite correct! But we need it if we're to record sound at all. Obviously, the loss of fidelity introduced by sampling techniques would reduce dramatically if the samples were taken at a very high speed. Similar is the case with sample size. If we use 8 bits of information per sample, we can have only 256 equal units with which to describe the signal. A 16 bits size would provide 65,537 such levels! So, why not go on increasing sampling rate and sizes indefinitely ? We can't, because this would also increase the amount of data we need to store per second of sound!

Digital sound storage parameters for consumer CDs were standardized some years back by International Standards Organization(ISO 10149), and these envisage a sampling rate of 44,100 times per second ("44.1 KiloHertz" or "44.1 kHz") and a sample size of 16 bits. With these the specifications were published. With these specs, uncompressed 2-channel stereo sound needs 1 MB of disk space to store around 11 seconds sound.  $(2(\text{channels}) * 44100(\text{sampling rate}) * 11(\text{seconds}))$ . This is clearly too much. There are ways of reducing this. First we have data compression techniques, and sound files are normally highly compressible, to around 8:1. Then we can use mono sound instead of stereo, thus cutting space requirements by half. Another common practice is to reduce the sampling rate from 44.2 KHz to 22.05 KHz. With 22.05 KHz sampling rate and 8 bits sample size voice quality as good as Yuvavani broadcasts you tune in every morning.

Undoubtedly the most exciting new technology for capture of information is voice input. Voice message systems now enable the storage, editing, and forwarding of spoken information. Voice information can be appended to textual

messages or files for distribution or to be keyed by a word-processing operator. More details can be found in Microsoft Sound System manual[24]. Because speaking is normally faster than typing, there is an ongoing market for dictation systems which send spoken words to operators for transcription.

### 1.4.3 Digitized sound vs Synthesized sound

Digital sound and synthetic speech are two entirely different kinds of sound. Briefly, digitized sound is recorded sound. To create digitized sound, you need an "analog-to-digital" converter. To make a recording, you speak directly into a microphone just as if your computer were a tape recorder. Since computers only work with numbers, the Voice converter (In our case it is Microsoft Sound Card) converts the sound energy to numbers. Sound card has A/D, short for analog-to-digital, which converts sounds into digital signal. The numbers are then saved in a binary file. To play the sound back, it's only necessary to convert the numbers back into sound with the help of D/A, short for digital-to-analog. Along with its other capability, sound card works as a D/A. Since digitized sound is recorded, it will sound natural. In fact sound card can play back digitized speech with a quality as good or even better than most cassette recorders. In contrast, synthetic sound is not recorded. It is created by the computer. With a bit of practice, you can edit synthetic speech to make it sound more natural, but synthetic speech always sounds a bit robotic.

Both types of speech have their advantages and disadvantages. Digitized speech sounds human and life-like, but must be recorded first. Since digitized speech must be stored, it can take lots of computer memory. Synthetic speech is machine made and sounds much less human than digitized speech. As an advantage, since the computer creates synthetic speech as it goes along, there's no

limit to how long it can "talk". It's not necessary to record the speech first. Text from the keyboard or a word processor may be "spoken" without having to record it first.

Synthetic speech has three major advantages over recorded speech :

1. No prior recording is necessary.
2. It can be used to "read" text, which makes it ideal for applications like talking educational software and the like.
3. Since there are no recorded speech files to be stored in memory, synthetic speech does not require as much memory.

Even very long text files may be "read". It's possible to "read" an entire novel aloud with synthetic speech. Provided, of course, the novel is on disk.

## **1.5 ORGANIZATION OF THE THESIS**

Chapter 2 outlines the survey of literature in this area. Even though acoustics of speech has been extensively studied for many years the speech application in an office environment and speech processing on computers are relatively new areas of research. Here survey of literature is limited to computer generation of speech (text to speech), multimedia applications where speech forms a component and voice messaging. Chapter 3 outlines the major feature of our work. Three representative prototype applications that have been developed as a part of this thesis are discussed. Chapter 4 outlines the conclusions and ideas for future work.

## CHAPTER 2

### LITERATURE SURVEY

#### 2.1 MULTIMEDIA

Considering the fact that speech can be produced with the help of computer, many new software with sound facility are coming up in the market. Also many multimedia application are emerging with sound and graphics capabilities. Recently Microsoft introduced device driver for PC speaker, which can produce voice as good as tape quality. There is no need of additional hardware for sound application except for recording.

Multimedia consists of an integrated whole which combines text, graphics, animation, sound, and video. Multimedia techniques revolve around a set of software and hardware tools which allow the creation of each of these five building blocks, and the ultimate combination of all these into a single spectacular entity. At the center of the multimedia development system is an "authoring tool", which allows the multimedia designer to build the structural skeleton of his application. Having designed his application, the developer can later fill out the specific components, either by building them from scratch using the individual creation tools, or by using ready-to-use sources such as clipart libraries, audio cassettes, CD-ROM etc.

Computer and software vendors have been struggling to add multimedia features graphics, sound, animation to personal computers. But for a long time multimedia has been more promise than reality, in part because we had no standards for this technology. The simultaneous introduction of the Microsoft-

endorsed Multimedia PC [15] and IBM's Ultimedia[15] strategy have finally brought consistent standards to this market.

The Multimedia PC (MPC) standard is based on a minimum specification for CD-ROM performance, sound capabilities, and display graphics, and is designed to run applications written for Microsoft Windows with Multimedia Extensions.

The IBM PS/2 Multimedia Model M57 SLC is a mix of excellent components. The machine has an enhanced 386SX processor, 16-bit audio from a new version of IBM's M-Audio Capture and Playback Adapter/A (ACPA/A), IBM's CD-ROM II with CD-ROM /XA (extended architecture), Digital Video Interactive (DVI) support, and an advanced internal speaker that eliminates the need for separate speakers. These features are added to the existing XGA graphics, built-in SCSI controller, and high-capacity 2.88 MB floppy disk drive found in IBM's PS/2 Model 57.

Just about everyone seems to be entering the multimedia field, often in various capacities. Traditional "content" producers, such as Hollywood studios and publishing houses, are joining consumer electronics and computer companies and entrepreneurial outfits as professional multimedia publishers. Digital Equipment is forming an education and entertainment division. IBM is publishing children's books. Microsoft is making movie guides; Lucas Films, CD ROMs for school children. Warner New Media is producing electronic titles. NBC is readying news for desktop computers; Knight-Ridder newspapers, a prototype "electronic daily."

What makes all this possible, of course, is rapidly evolving digital technology, and the efficiency it offers in manipulating, storing, and retrieving information. Affordability and performance levels are finally intersecting in the 1990s to make multimedia feasible for consumers.

## **2.2 TEXT-TO-SPEECH SYNTHESIS SYSTEM**

A general outline of a speech synthesizer is given in the form of flowchart of TTS system. This flowchart represents the complete process whereby the computer acts as an intermediary, reading text generated by a third party. The complete system consists of a text pre-processor, parsing and part of speech assignment algorithm stress, and intonation computation, pronunciation module, computation of the prosodic variables of intonation, timing, and loudness, the computation of synthesis parameters, and synthesis of waveform from the parameters. To make this process more meaningful, we describe each stage of the synthesis scheme in more detail.

### **2.2.1 Text Processing**

The process of text analysis converts the printed text into linguistic structures that establish the phrasal hierarchy of the text. The phrasal hierarchy is necessary to convey the message of the text to a listener. Without this hierarchical structure, all the words in a sentence would be of equal importance, and the speech would lack focal centers. Such speech would also be unnatural, since human speech employs this hierarchical structure. The phrasal hierarchy consists of a tree-like structure of smaller phrases that combine to make larger phrases and sentences. Each node in this hierarchical tree is assigned a different level of stress

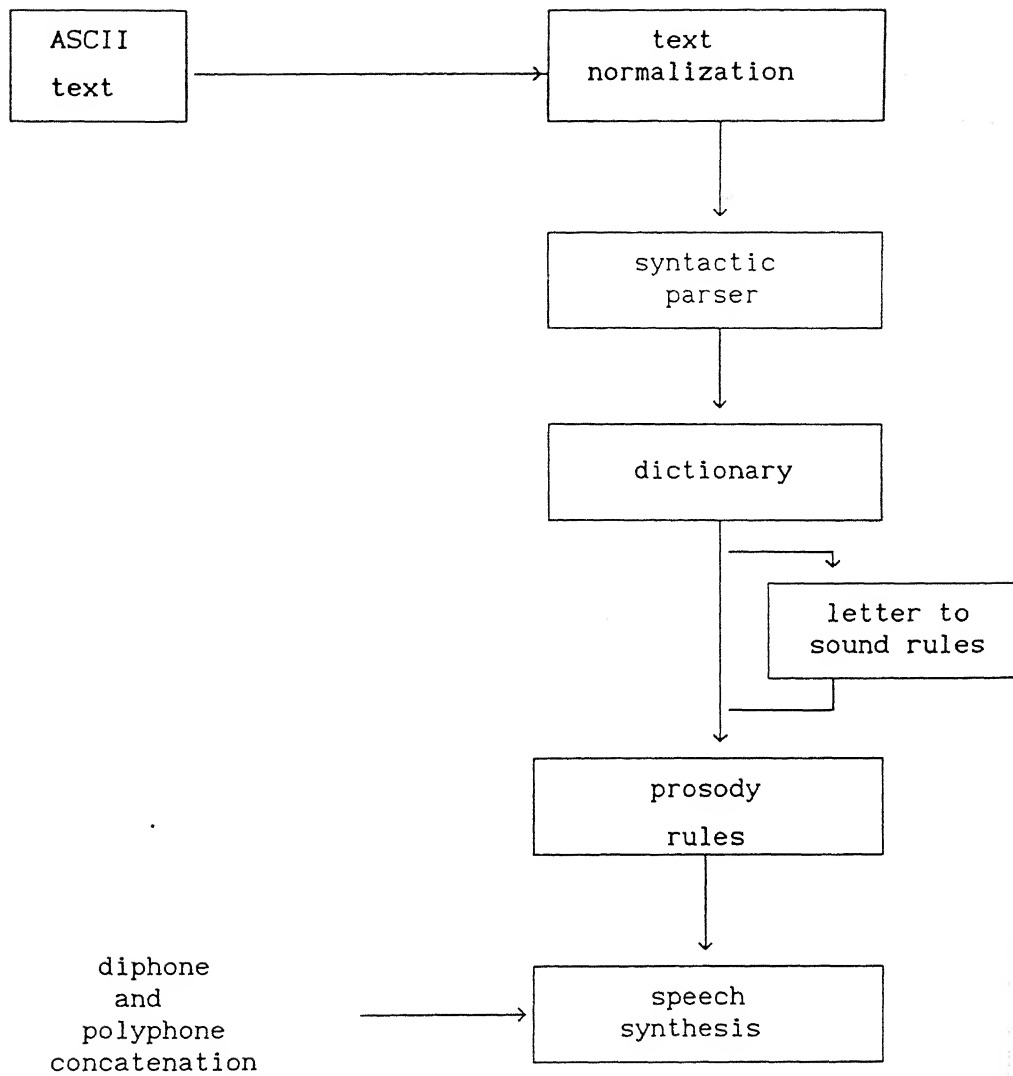


Fig 2.1 TTS Synthesizer



according to its importance, thus providing the listeners with information about the structure of the text. Phrasal structure is conveyed by intonation and duration variation; phrases are assigned different tone levels according to their stress level, and phrases are marked by pauses and proportional lengthening of words. Tones also differentiate phrase and sentence types (declaration, question, continuation, exclamation, etc.). In addition, phrasal hierarchy and intonation structure convey semantic information that is not governed by the syntax but by the intent of the text.

To perform a proper text analysis, a text understanding system is necessary. Present day technology cannot achieve text understanding; consequently, we must be satisfied with a more limited system of text analysis. Our text analysis is divided into three parts : 1. text analysis 2. Parsing and intonation and 3. letter-to-sound rules.

#### *2.2.1.1. Text Analysis*

For unrestricted text synthesis, we use a general text preprocessor to decipher numbers, special symbols, abbreviations and acronyms. The text preprocessor can cope with numbers in different contexts such as time, date, telephone numbers, and arithmetic abbreviations (e.g., dr for doctor and drive) and acronyms correctly.

The processor also assigns parts of speech to each word in the input. This parts-of-speech analysis is important for phrase focus, and properly pronouncing different words that have the same spelling. For limited tasks where the structure of the text is known, for example, electronic mail, news, items of inventory data,

the preprocessor can be tuned for the specific application to provide the proper linguistic structures.

This present system can cope with unrestricted text but, without understanding the text, the preprocessor cannot be expected to perform perfectly. The text preprocessor's interpretation of numbers and abbreviations and its assignment of parts of speech will produce some errors. To improve the text preprocessor, large amounts of varied text must be synthesized, and the algorithms changed to correct for preprocessing errors whenever they occur.

#### *2.2.1.2. Phrasing and Intonation*

After the text has been preprocessed, the synthesizer establishes the phrasal hierarchy of the text. Parsing text correctly requires a certain amount of text, understanding and, in the absence of that understanding, some compromises must be made. When the sentence structure is clear, the sentence can be parsed into the proper phrases, but it is advisable to break the sentence up into phrases only where the structure is clear. When a sentence is parsed incorrectly, the spoken text might convey a different message than was intended. For example, "Bill doesn't drink because he is unhappy" has two different meanings depending on what "because" refers to. The two cases are easily differentiated by intonation.

#### *2.2.1.3. Letter to Phoneme Conversion*

After the text has been parsed, stress values assigned to the proper words, and part of speech assigned to the word, the synthesizer has to know how to pronounce each word. Each character of the alphabet set used for the orthographic representation of text does not have a single sound associated with it.

Therefore, the alphabet characters must be transformed to a set of characters that describe pronunciation in a more direct way, namely a phonemic transcription. The process of text to phoneme conversion deals with the pronunciation of words and the pronunciation of proper names.

In this work researcher store a large dictionary and rely on letter-to-sound rules only when a word is not found in the dictionary. This method of converting text to phonemes yields an accuracy above 99 percent. The dictionary originally contained approximately 57,000 different words. However, many of the words in the dictionary are obtained from a common root by adding different prefixes and suffixes. Because of the work of Coker[20], the dictionary was condensed to about 30,000 words, and the system can generate 166,000 words.

### **2.2.2 Interpretation of Linguistic Structures**

The analyzed text is represented by phonetic characters, stress value, minor-and major-phrase markers, and intonation descriptors. The synthesizer uses this information to compute a speech signal in several stages. First we compute the duration of the different speech events, next we convert the intonation descriptors to a fundamental frequency contour, and then we generate loudness control. After these prosodic parameters have been computed. we generate the synthesis parameters that describe the different sounds or phonemes. These parameters are converted to speech by a waveform synthesizer.

The first step in the interpretation of the linguistic structures of speech consists of describing the timing associated with different events of the speech. More specifically, we need to compute the duration of the different phonemes. Each phoneme has an intrinsic duration however, in context these durations are

changed according to numerous factors, e.g., syllables, the position of the phoneme in the syllable word and phrase and the speaking rate. In our particular system, We have encode the rules that govern the duration of the phonemes in the form of a table. The rules are written in a simple linguistic notation, and thus are easily modified.

The phrasing and intonation computation produce a phrase-dependent intonation description that consists of a set of tones for each phrase and stress unit. The present system uses these tones to compute a fundamental frequency contour. These tones are time-aligned to the primary stressed syllables in the text. The fundamental frequency contour is obtained by interpolation between these tones and low-pass filtering the resultant contour.

Only segmental amplitude variations are used in this synthesizer thus far, but we believe a certain amount of loudness control over syllabic units would enhance the quality as well as help differentiate between stressed and unstressed syllables. This effect has not been studied yet, because we believe improvements derived from loudness control would not be as important as other effects in the synthesizer.

Once text has been transformed into phonemes and their associated duration, and a fundamental frequency contour is available, the system is ready to compute the speech parameters for text synthesis. In this case, researchers chose LPC as the parametric representation. We use a concatenate approach to compute parameters for the speech. The parameters are derived from tokens of natural speech and stored as transitions between certain phonemes or groups of phonemes.

Pitch adjustments are only necessary for voiced sounds. A frame of parameters flagged as voiceless is output to the waveform synthesizer without any alterations. For voiced frames, the length of the excitation is adjusted to be equal to the period previously computed. If the excitation is to be shortened, an equal number of samples is deleted from the beginning and end of the period. To keep the timing of the utterance and the individual entities within the utterance, a frame of parameters is dropped or repeated whenever the time of the output parameters is different from the input parameters by more than half of a pitch period. After the parameters have been set with the new pitch period they are sent to the waveform synthesizer. The final stage is a text-to-speech (TTS) that convert the analyzed text to speech. Parameteric synthesis is simple but computationally intensive.

Lopez[12] refers a text-to-speech system for Spanish with a large frequency domain. Bigorgne et al [4] present a multilingual text-to-speech that has been developed for German, Italian and English language. It has an ability to provide naturalness to the speech also. Idiap , C.P. [11] presents a grapheme to phoneme based text-to-speech that provide a very high level of accuracy.

It has been found that some words are used more often than others. TTS can be implemented with the help of dictionary. Mannell and Clark[13] outline in their text-to-speech(TTS) has been implemented with the help of dictionary. It has been developed around a lexicon knowledge base which contains the 4000-5000 most common English words and which has been augmented by a suffix stripper and a set of grapheme to phoneme rules. Evaluation and development of this system has been facilitated by using weighted statistics which reflect the frequency of occurrence of each word.

Some hardware products are also available for TTS. One of them is DEC product. DSC-2000[6] is Speech system designed for text-to-speech conversion. Digital Sound Corp moved closer to integrated voice/text messaging with its DSC-200 Voice server 1 store and forward system. Presently it is configured to handle voice only, although its Motorola 68000-based general-purpose processor and dual Texas Instruments' TM320 digital signal coprocessors can be programmed to handle text to speech conversion or text alone. Future plans call for incorporating limited speech recognition capabilities for commands and user access, as well as Unix support for any private branch exchange(PBX) based networks.

Such high level integration is beyond the capabilities of current message store and forward systems. Typically, separate systems are needed for voice store and forward and for electronic mail. Common user interfaces tie the two together at the application level. For example, Digital Equipment Corp's DEC-talk II text to speech system can be combined it with a voice messaging system from Voice Mail International.

Still, microprocessor-based voice managing systems tax the limits of speech technology. A problem develops because of contradictory goals. Low overall costs can be achieved only if microprocessors handle the same functions as mainframes. On the other hand, there must be significant digital signal processing to bring sample rates down to 16 kbits/s or less. This will allow microprocessors to process voice information as just another data type.

Unfortunately, there are no easy solutions. Mathematical models, which serve as the voice digitization/reconstruction foundation, fall apart when sample

rates go outside their set thresholds. Due to the more general approximation of sampled speech, voice quality deteriorates as noise components are introduced.

## CHAPTER 3

### DESIGN AND IMPLEMENTATION

#### **3.1 INTRODUCTION**

Based on the survey of current development elsewhere and taking into consideration the Indian environment we have developed in this thesis some representative applications of sound in an office environment. The applications have been chosen as to represent a wide spectrum, and sufficiently general in nature so that they can be adopted to suit a variety of situations. All the applications developed use minimal hardware, software and avoid 'looking into' any proprietary hardware/software. These applications do represent state of the art techniques and use dominant standards in hardware and software. The applications have been implemented completely so as to benefit potential users of their applicability.

#### **3.2 SCOPE**

In this thesis we are primarily interested in demonstrating the application of speech in an office environment. Our aim is to demonstrate some interesting applications for Indian environment; low cost and versatility would characterize an application environment.

Possible sound applications are low-end, medium or demanding application. As a matter of fact the low-end application like Bajikhaye's[3] work which was done using Speech Thing[23] is hardware dependent. It is also a non-generic type program. A high-end sound application requires expensive hardware platform like an engineering workstation and many man-months of efforts.



thesis. We outline three prototype applications that represent a mix and innovative efforts.

1. Text-to-speech converter for Devanagari : In this work a software that reads input text in an Indian language script from file and generate voice output through a speaker is developed.

2. Voice Query : In this application the result of a database query is transformed into a voice output.

3. Voice Announcement System : In this application a text file is converted into voice output.

These applications are "prototype applications"; they can be extended to a wide variety of situations applications to suit the user requirement.

### **3.3 DESIGN ISSUES**

In the previous chapter many aspects of voice and the voice applications have been discussed. The application includes multimedia, voice mail and messaging systems.

In this thesis we limit ourselves to PC based hardware platform to reduce costs and increase versatility in the Indian environment. Other decisions that

affect an design are discussed below. Our aim in this thesis is to demonstrate voice application in the field of office automation. Our experiment initially used a commercial product Interactive Sound System . It is capable of storing and playing digitized (recorded) sound and transmit it over standard PC based network. The major drawback of this system is the lack of programmability. Since our interest is of developmental nature we were looking into sound cards/Sound Software products that would provide programmability. With this in view we used MS-Sound[24] in this thesis.

MS Sound is versatile and flexible as it can store and play various qualities of sound ranging from tape quality to CD quality. It can add effects like Normalize, Fade in/out, Echo , Volume, Speed, Filter out, Trim out etc. , on sound files. It has an additional feature of OLE (object linking and embedding). If one is working in a document created with an application that supports object linking and embedding, one can add a sound object using MS-Sound System. For example with the help of OLE one can paste announcement in e-mail. If one has compact disc read-only memory(CD-ROM) drive connected to one's computer, one can play audio compact discs (CDs) on the drive with Music Box of Sound System. It can play different formats of sound files. One can also convert file from .AIF, .SND and .VOC format to the .WAV format [24], which is Microsoft standard for representing digitized sound. It is increasingly becoming the dominant standard widely used in industry. It has also low-end voice recognition known as voice pilot [24]. It can be used to issue commands to program through speech. It has in-built vocabulary to recognize some of the common command words of standard PC software and MS DOS operating System. One can re-train the words as well as add new words to vocabulary. It has low-end text-to-speech conversion for numeric data and interfaces with MS-Excel[25] spread-sheet product. Even

though MS-Sound System is versatile, it also lacks full programmability, we use it for recording but for use in programs we resort to a sound library from TegoSoft [14].

This library has `sp_` and `ts_` function for playing and recording sound. The `sp_` functions are primarily for playing sound files on PC-speaker while the `ts_` functions are for recording and playing sound files with the help of sound card. There are two major differences between playing sound files through the built-in speaker and playing through a sound card. To begin with, the sound card produces better sound quality. Secondly, the sound card does not use the CPU resources for sound playing. This means that once the CPU issues a command to the sound card to play a sound section, the sound card is on its own, playing the sound section without any help from the PC's CPU. The PC is free to perform other tasks while the playback is going on.

We use Borland C++ as our software platform. The reason why we selected this language is that many sound library & utilities are available for C/C++ & Windows programming. Moreover C/C++ language are defacto standard for many multimedia application. Recently Microsoft Windows introduced device drive for PC speaker, and it may be possible in future that Microsoft provide some sound utilities.

Microsoft Windows has been around since 1985. This is remarkable because the computer hardware needed to effectively run Windows did not become broadly available until about 1988. Microsoft had the vision to continue the development of Windows, and Windows application programs like Excel[25],

long before the market developed. This vision paid off in 1990, with the tremendous success of Windows 3.0.

At the time of writing Windows has become "the" programming environment for the PC family of computers. This is great for PC user, but demands a lot from programmers. Besides needing to know how to use a programming language such as C or C++, the novice programmer is confronted with the Windows API (Windows Application Interface) with over 600 functions, 200 messages, and a wide range of unfamiliar terms and concepts. TTS make matters worse, Window programs are structured completely differently from conventional programs that run under DOS. Despite these obstacles, Windows programs are not difficult to create, once the basic principle are understood.

Although other languages can be used for Windows programming, Windows development today is done almost exclusively with the C and C++ programming languages. In the work which follows the reader is assumed to be familiar with the C language, but not an expert.

**Hardware Requirement for Windows:** To develop Windows programs, you need a computer that will run Windows. The minimum system is an 80286 based PC with 2 megabytes of memory, a hard disk, and an EGA or VGA video screen. Faster computer with more memory speed up the things, but they are not a necessity. A mouse is a necessity to use several of the programming tools.

**Software Requirement for Windows program:.** A copy of Windows 3.0 or 3.1 should be installed on machine. For programming, machine needs a C compiler that supports Windows and a Windows resource compiler. In the past, this meant

buying a DOS-based C compiler and a Windows resource tool kit separately. Both Borland and Microsoft now sell the compiler and resource tools as single package, including Windows-based development tools. These new tools are ideal for learning to program Windows, as many of the low-level details are handled automatically by the programming tools, freeing the programmer to concentrate on creating programs.

Both Microsoft and Borland offer advanced versions of their compilers, called Microsoft C++ and Borland C++. These tools include "command line" compilers. When creating a Windows application the command line compilers usually are run by opening up a DOS window from within Windows. One advantage of the command line compilers is that they allow more complete control over the compiler's code generation and optimization features than is available in the IDEs. These features are most important in large projects, and in cases where optimizing the speed or size of the program is critical.

Dynamic Link Libraries, or "DLLs" for short, provide groups of function for Windows applications to use. One DLL can be accessed by any number of applications at the same time. DLLs make efficient use of memory if the same functions are needed by several applications, because only one copy of the code is needed. For the programmer, DLLs provide the ultimate in reusable code. Once a DLL is compiled and debugged, it never needs to be compiled again or linked into another program. DLLs become an extension of Windows environment, adding new functions to those already provides in Windows. Windows itself can be thought of as a collection of DLLs, as all of the code for functions (such as `CreateWindows()` and `TextOut()` ) reside in DLLs (such as `KERNEL` and `GDI`).

The DLL file resides on the hard disk and is shared by all the programs that use it. Because the DLL is not part of the executable programs, ones program's size decrease. Suppose ones application is composed of several programs, and each program calls sp\_ functions. In such a case, it is probably better to utilize the DLL. Another advantage of using a DLL is that it can be used from other Windows programming languages, such as Visual Basic for Windows. It can also be used from any other Windows programming language that can utilize DLL.

The disadvantage of using a DLL is that ones application assumes that the end user has the DLL installed on the hard disk. Because most users do not have the DLL sound library, ones distribution disk must include the DLL, and your Install program must copy the DLL into the user's hard disk.

Our software used TegoSND.DLL library[14]. We can convert an application that can use static library. TegoWin.lib is Static library provided by TegoSoft. By using Static library we can develop a stand-alone application.

Following steps are required to make it stand-alone application with static library.

1. Replace statement `#include "c:\spSDK\TegoWlib\sp4Win.h"` with the statement `#include "c:\spSDK\DLL\TegoSND.h"`.
2. The application uses sp\_ functions from the DLL. Therefore, the .DEF file must contain the appropriate `IMPORTS` statement. This `IMPORTS` statement

contains all the sp\_ functions that the application imports from the DLL. So following statements are to be included in .DEF file.

## IMPORTS

TegoSND.sp\_OpenSession

TegoSND.sp\_PlayF

etc.

3. In the tlink statement in .bmh library name should be change.

## 3.4 IMPLEMENTATION

### 3.4.1 Introduction

In this thesis we implement three tasks to illustrate the power of voice in an office environment. The implemented task have been so chosen as to represent an interesting spectrum of application demonstrating the different facets of voice.

In section 3.4.2 we detail our implementation of an elementary Text-To-Speech(TTS) scheme for Devanagari script (a script used by the Official Language of India, viz Hindi Language) derived from the ancient Sanskrit Language. In section 3.4.3 we outline an application that is less demanding but of wider appeal in contexts of announcement, annunciation, alarm etc. Section 3.4.4 details a more demanding application that interfaces voice with a database look up. Using imaginative schemes these prototype applications can model a wide variety of interesting and innovative situations to enrich office productivity.

### 3.4.2 Application A: Devanagari TTS

Chap 2 outlines the technology of TTS and the different attempts to solve the general problem of TTS by several researchers. Text-to-speech is implemented with the help of dictionary in which large number of words are recorded in different verb-forms separately. It can also be phoneme-based speech. Every language has its own basic phonemes. Most of the languages have 30-35 phonemes. Any word is combination of several phonemes. In phonemes-based TTS word is produced by concatenating phonemes. Phonemes-based TTS can be digitized text-to-speech or synthesized text-to-speech. In synthesized text-to-speech phonemes are stored in computers.

In digitized text-to-speech we have to store / record phonemes. Unlike Indian languages are phonetic by nature. In synthesized text-to-speech words are produced by computer. It is suitable to language like English which is not phonetic by nature. In such language pronunciation of characters depends upon its context. Expansion of some English words into phonemes are given below[23].

Cat	kAEt
Computer	kAXmpyUWtER
Smooth Talker	smUWDHtAAkER
First Byte	fERstbAYt
Software developer	sAAftwEHrdEHvEHlAXpER
Pascal	pAEskAEI
Comdex	kAAmdEHks



## Some of standard phonemic transcriptions

AE Short "a" as in "last".

EH Short "e" as in "best".

IH Short "i" as in "fit".

AA Short "o" as in "cot".

Indian languages are phonetic by nature. They have their own advantages and disadvantages. Advantages are that we have to write simple algorithm to separate the phonemes of words. But to record Halant(half character in Indian language) is difficult and other English & European languages do not have any half characters like our Indian languages. Moreover pronunciation of the word is not exactly the combination of individual phonemes in Indian languages.

Our implementation use digitized speech for single character and sound for a word is generated by concentrating the digitized sound for the individual characters that make up the word. In this thesis we use an innovative scheme that is specially useful for Indian languages that explores the phonetic nature of Indian language scripts [16]. While the technology to handle Indian language scripts is available in the form of GIST[26], we are using a simplified scheme designed by Sangal and Chaitanya[16] with an extension to suit our needs. To denote the 'Halant' character of Devanagri we use a postbox 'L' after the equivalent consonant character. This translation will be refereed to as D-E Scheme (Devanagri-English).

Our implementation roughly follows the following scheme

Step 1. Pre processing of Devanagari script to English script (done by the user).

Step 2. Parsing of Devanagari script to an D-E scheme.

Step 3. Converting the D-E Scheme string to speech output.

*parser.c* program accomplishes step 2. The complete Devanagari script as it is generally used appears in Table 1. This table also shown the name of the file that contains the sound corresponding to this symbol. The filenames follows D\_E Scheme with the modification that the upper case character are duplicated for example xA in DE Scheme is store in xAA.WAV file. This was necessary as DOS handles lower and upper case character as equivalent. The word separator is a set of double spaces.

*tts.c* program produces the sound output corresponding to the input string generated by *parser.c*. It concatenates all the character of a word and gives the sound output to the PC speaker. The PC speaker is driven by TegoSoft.sp\_Play function. Some of other functions are also provided by Windows & Borland compilers. Borland C++[14] has OpenSound(),StartSound() and StopSound() functions for sounds. To remove the glitch, in combining sound files we remove tail(around 1 Kbytes) of first files and remove header(40 Bytes) and tail (around 1 Kbytes) of remaining files.

Caution : We can combine all words in a single temp.wav file, but in that case temp.wav file will be very large in size. Due to large size of sound file TSEngine has to load data in memory chunk by chunk. So there will be small break each time in playing sound file after few seconds.

Basic components needed to execute the software are as follows.

1. Microsoft Windows.
2. TTS.EXE file.
3. TEXT.DOC file in which input text is there.
4. Wav files corresponding to phonemes.

Limitation : The pronunciation of a word is more than just combination of sounds of the individual alphabets that together constitute the word. Due to this reason the sound produced does not sound natural. The smoothness of human speech is also missing in this scheme.

### **3.4.3 Application B: Voice output**

In this section we implement a simple application that adds voice output to a text file. Typical uses of this prototype application are announcement system regarding Arrival/Departures/Delays of trains/aircraft/bus in a Railway station or Airport or a bus station. Such automated system permits flexible, announcement whose sound quality can be time invariant. (As it happens with human operators). The implementation is a significant improvement over Bajikhaye's[3] that used a proprietary hardware from Covox[23]. Our implementation works on any PC with no additional hardware.

Advantages of voice in announcement system :- In announcement mostly same information is to be played more than once. Due to repetition of work, announcer gets tired to speak same thing. The tone of announcement is also depends upon person's mood. In voice system we do not need announcer. Only

at the time of any change in schedule, administrator has to change information text. This system can be integrated with phone thereby extending the scope of potential application.

For our application we need all words on hand before announcement is to be played. We have to store all train numbers, train names, platform numbers and times in a computer. These speech files are stored in digital form. Computer sends the files to speaker which produces speech. We can also use sound card for this purpose. With the help of sound card, speech quality improves. We can take also send output of sound card to speakers with amplifier and audible to many passengers. In an implementation we have both the options available.

Limitation: The current version has in its database the following train details platform numbers are limited to 10 and timings are rendered to an hours(0-24) to optimize the space requirement as the WAV files occupy considerable space. We use the sound of 1-12 to generate all 24 hours by use of A and P prefix. We compose the full sentence before announcement as it takes time to open the WAV file sessions.

#### **3.4.4 Application C: Voice Query**

This application combines the database function with voice announcement and represent a medium range application. A variety of interesting application can be built using this prototype in banking, insurance, finance and marketing sectors.

The illustrative example is a Banking Announcement System from a database that contains the following details A/C No and Balance. The balance

being a decimal number digits with the digits, following decimal point representing paise. We use this knowledge to generate the voice output with minimum efficiency. We do not convert the number into place value system but read off digit by digit with the appropriate configuration of Rupees and Paise. This minimize substantially the requirement to store a large .WAV file. All that we need is to store sounds corresponds to digits 0,1,..9 (approximately 9 Kbytes per digit).

Typical uses of this prototype application are query system regarding the read out of bank balance of an input account number. Such query system permits flexibility as query can be made at any time irrespective of whether bank is open or not.

For our application we are taking account number from file. We can extend it to take it from menu. By interfacing with phone we can take input over telephone line and reply on phone. We can also use sound card instead of PC speaker. With the help of sound card, speech quality improves.

Limitation: The current application is only speaking out account number and its balance. We stored only sound file of digits (0-9) to optimize the space requirement as the WAV files occupy considerable space. We use the sound of 0-9 to generate all numbers.

### **3.5 LIMITATION**

As we developed software for PC-Speaker sound quality is not as good as business quality. Moreover CPU, itself is playing sound section. The Software

can be executed on any other computer but clock speed should be high. Software was developed for PC platform, so it can't work on any other platform.

For PC-Speaker sound quality can be of tape quality, stereo quality or CD-quality. Secondly, sound file for PC has sample size of 8 bits, so PC can't play sound section whose sample size any other than 8 bits (like CD-quality with 16 bits sample size). It can only play .WAV format files.

We are generating word by combining different phonemes, so size of sound file depends on number of phonemes in the word. Sound data is sampled data of analog signal of sound. So sound file data is the record of continuous signal. The pronunciation of a word is more often than not an perfect combination of the phonemes of the individual alphabets that together constitute the word. In combination of different phonemes, produced word gives small glitch between two phonemes. Many other researchers also obtained the same result.

We are presently using standard .wav format. We are not using any compression/decompression technique to store data. In compression / decompression we have to save memory at the cost of performance.

# CONCLUSIONS AND RECOMMENDATIONS

## 4.1 CONCLUSIONS

This thesis has attempted to build some generic applications of voice in the office automation context. Keeping in view the price sensitive Indian scenario the applications are low cost implementations. To provide versatility the implementations are very modest in their demand of hardware and software resources. To provide a user-friendly interface extensive use has been made of the Windows [10] Graphical User Interface. To provide for software stability most of the code is implemented on Borland C++[14] development environment. The development of applications need a sound card to digitize sound; running the application does not require the sound card and thus provides maximum flexibility.

Voice applications being very demanding, generally were attempted on high cost engineering workstation only. This thesis attempts to provides such demanding applications even in a low cost PC environment.

The text-to-speech conversion that demands very extensive effort for European languages lends itself to elegant solutions with phonetic script of Indian languages. With the pioneering work done by C-DAC through their GIST technology inputting Indian script through keyboard has been greatly simplified. With this added facility, our text-to-speech for Indian languages script assume special significance. The voice output for textual database and to the database query opens up several avenues for interesting applications.

## **4.2 RECOMMENDATIONS FOR FUTURE WORK.**

Due to limitations of time and the fact that most of the effort initiated in this thesis has been attempted for the first time we could not fine tune the implementation completely. While the phonetic nature of the Indian language scripts provide for straight forward synthesis of sound from the character representation, one needs to fine tune our work to provide continuity, smoothness and naturalness. Sound file being very large need time to be read by the processor before being output to the PC speaker causing delay that may not always be acceptable. Better compression schemes and improved processor speeds would address this problem. Our database query could significantly benefit by interfacing the PC to a telephone set so that any user with a telephone access can get the database access.



## REFERENCE

1. Adam, John A., "Interactive Multimedia", IEEE Spectrum, March 1993, pp-22.
2. Adam, John A., "Applications, implications: Special Report/Multimedia" , IEEE Spectrum, March 1993, pp. 24-31.
3. Bhajikhaye, R., "Voice in office automation implementation of a voice response system applicatoin", M.Tech Thesis submitted to the IME Dept, I.I.T.Kanpur, 1993
4. Bigorgne, D., "MULTILINGUAL PSOLA TEXT-TO-SPEECH SYSTEM" ICASSP (CD-ROM) - IEEE International Conference on acoustics, speech, signal processing contains information on subjects and related papers, 1992-93.
5. Bajaj, K. K., "Office automation", Macmillan Series in Computer Science, Macmillan India, 1st Edition, 1989.
6. Centigram Corp, Octel Communication Corp & Digital Sound Corp. "Speech system designed for text-to-speech conversion", Computer Design, December 1984. pp 126-127
7. Conger, James, "Windows API Bible", Galgotia Publication, First Edition, 1993, pp. 741-757.
8. D'alleyrand, Marc, "Data Capture Technologies" in Image Storage & Retrieval Sysem, McGraw Hill, 1989, pp . 65-99.

10. Conger , James, "Windows Programming Primer Plus" ,Galgotia Publication First Edition, 1993.
11. Idiap, C. , "An Efficient Way to Learn English Grapheme-to-Phoneme Rules.", ICASSP (CD-ROM) - IEEE International Conference on acoustics, speech, signal processing contains information on subjects and related papers, 1992-93.
12. Lopez ,Gonzalo et. al. , " A TEXT-TO-SPEECH SYSTEM FOR SPANISH WITH A FREQUENCY DOMAIN BASED PROSODIC MODIFICATION.", ICASSP (CD-ROM) - IEEE International Conference on acoustics, speech, signal processing contains information on subjects and related papers, 1992-93.
13. Mannell, R. and Clerk, J.E., "Text-to-speech rule and dictionary development", Speech Communication, Vol. 6. No 4, December 1987, pp pp317-322.
14. Mukhi, Vijay , " Borland C++ 3.0 for Windows 3.1", BPB Publication, 1993.
15. Gurewich, Nathan & Gurewich, Ori, "Programming Sound for DOS and Windows", SAMS Publishing, First Edition, 1993.
16. Raskin, Robin, "Is Multimedia Real?", PC Magazine, Dec 17, 1991, pp. 35-46.
- 17 Sangal, R. and Chaitanya, V., Devanagiri English Transliteration Scheme, Personal Communications (Jan 1994).

18. Shenwai, S. and Kumar, D., "Multimedia Beyond the Madness", C&C, Nov. 1993, p.p. 124-128.
19. Reisman, S. , "Birth of a business : Multimedia gets real", IEEE Software, May 1993, pp-122.
20. Stephen E., Josesph, P., and Judith S., "Speech Synthesis in Telecommunications", IEEE Communication Magazine, November 1993.
21. Tapscott, Don, "Office Automation A User-Driven Method", Plenum Press, 1982.
22. Centigram Corp, Octel Communication Corp & Digital Sound Corp. "Speech system designed for text-to-speech conversion", Computer Design, December 1984.
23. User's Manual, Speech Thing Covox Inc, Sixth Edition, May 1991.
24. User's Guide, Microsoft Windows Sound System, Microsoft Corporation, 1993.
25. User's Guide, Microsoft Excel, Microsoft Corporation, 1990.
26. User's Manual for GIST ASP, Centre for Developement of Advanced Computing, Pune (Jan 1993)